Description of a Few e-HTPX Use Cases

Graeme Winter
1st February 2005
Version 0.3

Assumptions:

Just to make the context of these use cases clear, the following assumptions have been made.

- The beam time has already been allocated and scheduled, so that the time and date of performing the experiment/measurements is known
- A member of the lab will be available on the beam line to perform manual tasks for example sample handling
- An experienced crystallographer "at home" will be in control of the experiment and it is this person who owns the crystals and wants the data
- Safety issues have already been sorted

Use Case 3.5: Automated Processing

To: Provide a service which will automatically process data, including indexing, integration and scaling.

This is the data processing which could occur as a part of use case 3 above, but could also be considered separately if desired. The input to this should be the images measured for instance in 3, and optionally some information which could help in the processing, for instance the correct space group, the number of residues in the molecule and so on. However, this process can be performed with almost no input information.

In the event that the meta data in the image headers is incorrect, the correct information must be supplied.

Inputs:
A substantial number of images.
Appropriate meta data from the experiment.

Outputs:
The refined unit cell. (as above)
Graphs of: (all are lists of ordered pairs float, float)
    Rmerge vs. batch.
    Mosaic spread vs. batch.
    I/sigma vs. batch.
    Completeness vs. resolution.
    Multiplicity vs. resolution.
    Intensity vs. resolution.
Determined point group. (string, from list)
Rmerge. (float, percentage)
Radiation damage analysis. (batch number, integer)
Strength of anomalous signal. (float, unit as yet to be determined)
Total number of reflections. (integer)
Total number of "bad" reflections. (integer)
Estimated B factor. (float)
Likelihood of twinning. (float)
Solvent content. (float, percentage)
Number of molecules / asymmetric unit. (integer)
Reflection files:
    Unmerged, unscaled.

Unmerged, scaled.
Merged, scaled.
Merged, scaled, converted to structure factors.

Use Case 5: Phasing *via* SAD

To: Provide a service which will take as input processed data in any acceptable form (e.g. unmerged intensities, merged intensities or merged structure factor amplitudes) along with information about the number of heavy atoms to find, and attempt to phase the data.

This is a more complex use case, since it is possible that this will fail. The input should be in the form of a file containing the processed reflections, and some meta data including:

- Unit cell
- Number and type of heavy atoms
- Spacegroup

This information will then be used to locate the heavy atoms, and these locations will form a part of the result of this process. Once the heavy atoms have been located, these should be used to calculate experimental phases for all reflections. The locations of the heavy atoms should then be refined along with the phases. Finally the phases may be refined for instance by density modification before returning the resulting phased reflections.

Inputs:
Reflection file from 3.5 or similar.
HA types & number.

Outputs:
Status.
Phased reflection file.
HA locations (x, y, z, occupancy)
Quality statistics (correlation coefficients etc.)

Use Case 6: Phasing *via* MR

To: Provide a service which will take as input processed data, a sequence of the protein, and a model to use for molecular replacement, and to use these to calculate phases for the reflections.

This is again a more complex use case. The input should consist of

- Unit cell
- Spacegroup
- Number of copies of the model to place
- The structure of the model

This information will then be used to try and place the correct number of copies of the model structure in the asymmetric unit. The transformations to perform this placement should be returned. These should then be used to calculate initial phases for the input reflections. Finally the model or number of copies of the molecule should be refined against the input reflections. The result of this process will be the new model, the new reflection file, and the orientations of the molecules. In addition some element of data improvement by density modification could be included.

Inputs:
Reflection file from 3.5 or similar.
Model to use for molecular replacement. (pdb file)

Sequence of protein. (.pir format or similar)
Number of molecules to place. (integer)

Outputs:
Status.
Number of molecules placed. (integer)
Orientation and position of placed molecules. (4 x 3 augmented matrix)
New model made from input model and transformations. (pdb file)
Quality metrics of placement, e.g. clashing etc. (yet to be determined)

Common thoughts.

In both cases consideration must be given for integration with existing and developing databases and data management systems, and it should be ensured as far as possible that time and effort are not wasted duplicating functionality which already exists, for instance in ISPyB.